

# ISIS FIE3 - Technologies du Web - le langage Javascript – API & AJAX

Mini-projet 2020-2021

Gestion du frigo

Jean-Marie Pecatte

Pour faire une gestion partagée d'un frigo (plusieurs utilisateurs du même frigo), il est nécessaire de gérer une base de données commune avec un backend coté serveur. La bonne pratique aujourd'hui c'est de donner accès à ce backend à travers une API.

C'est ce que l'on va faire avec notre frigo. Mais comme l'objet du cours est uniquement le coté client, l'API de gestion des données vous est fournie.

Elle est accessible à cette URL :

<https://webmmi.iut-tlse3.fr/~jean-marie.pecatte/frigo/public/{idfrigo}/produits>

L'API gère plusieurs frigos ; {idfrigo} est l'identifiant du frigo ; pour le frigo 1, l'url sera donc :

<https://webmmi.iut-tlse3.fr/~jean-marie.pecatte/frigo/public/1/produits>

Un produit est caractérisé par un id, un nom, une qte.

Les quatre opérations classiques (Create, Read, Update, Delete) sont possibles sur un frigo :

- Lister les produits du frigo
  - o GET URL -> un tableau de produits (au format JSON)
- Ajouter un produit au frigo
  - o POST URL avec en data un « produit » (au format JSON sans id) -> { "status" : 1 ou 0 }
- Supprimer un produit du frigo
  - o DELETE URL/{id du produit} -> { "status" : 1 ou 0 }
- Modifier un produit du frigo
  - o PUT URL avec en data le « produit » modifié (au format JSON) -> { "status" : 1 ou 0 }
- Une autre opération permet de rechercher des produits à partir d'un mot clé :
  - o GET URL?search=XXX -> un tableau de produits (au format JSON) qui contiennent XXX dans leur nom
- Une autre opération permet de récupérer les infos d'un produit à partir de son id
  - o GET URL/{id du produit} -> le produit (au format JSON) s'il existe ou { "status" : 0 } sinon

---

➔ Écrire une application HTML/CSS/Javascript permettant de gérer un frigo :

- Afficher le contenu du frigo
- Ajouter un nouveau produit au frigo { *nomproduit*, *qte* }
- Supprimer un produit du frigo [*totale*ment *quelle* que soit la *qte*]
- Ajouter 1 à la qte d'un produit présent dans le frigo
- Supprimer 1 à la qte d'un produit présent dans le frigo (si la qte est nulle alors supprimer le produit)
- Rechercher des produits dans le frigo

Contraintes :

- Chacun de vous utilisera un frigo différent ; utiliser l' {idfrigo} correspondant à votre rang dans la liste d'appel ;
- La mise en forme HTML/CSS sera aussi évaluée (cf. cours de Mme Pecatte) ;
- Vous devez mettre le code de votre mini-projet dans votre workspace dans codesandbox

- Vous ne pouvez pas utiliser de librairie ou framework CSS ;
- Vous ne pouvez pas utiliser de librairie ou framework JS ;
- Votre code doit respecter les bonnes pratiques de dev (clean code) ;
- Il existe aujourd'hui des outils permettant de comparer deux programmes et d'obtenir une estimation de leur similitude ; s'il s'avère que votre code est « similaire » à l'un de vos collègues de promo, la note de votre mini-projet sera proche de 0...
- Date limite de rendu : vendredi 09/04 18h dernier délai ;
- Déposer le lien de votre codesandbox dans le devoir prévu sur moodle.

---

### Quelques éléments de code :

Pour faire des requêtes POST, DELETE, PUT

→ il faut changer la valeur de la propriété method des options du fetch ;

De même, l'envoi des données vers le serveur (POST & PUT) se fait via le corps (body) de la requête http ;

→ à mettre aussi dans les options du fetch

Exemple de requête POST :

```
let myHeaders = new Headers();
myHeaders.append("Content-Type", "application/json");
const fetchOptions = {
  method: 'POST',
  headers: myHeaders,
  body: JSON.stringify(produit)
}
fetch(url, fetchOptions)
.then( (response) => {
  return response.json()
})
.then( (dataJSON) => {

})
.catch( (error) => console.log(error))
```